REMARKS/ARGUMENTS

Claim 1 has been amended to recite that a system handler of an operating system is operable to dynamically link with a plurality of gaming program shared objects and device handlers for the computerized wagering game at run time when the computerized wagering game is executed in a manner that allows the plurality of gaming program objects to call a set of common functions effectively provided by the system handler application when the system handler application is executed and load said gaming program shared objects and device handlers (see, for example, page 8, lines 22-23 and page 9, lines 14-16 of the specification). It is respectfully submitted that the cited art does not teach or suggest this claimed feature.

In the Final Office Action, the Examiner has rejected claims 58-68 and 71-75 under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 5,742,825 (*Mathur et al.*). In addition, the Examiner has also rejected claims 69-70 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 4,757,505 (*Marrington et al.*). The Examiner's rejection of claims is fully traversed below for at least the following reasons:

(a) *Mathur et al.* does not teach or suggest: a system handler application of an operating system, wherein the system handler application is operable to load a gaming program shared object in response to a change in the stored game data variables by another gaming program shared object (claim 1).

In the Final Office Action, the Examiner has asserted that *Mathur et al.* teaches this claimed feature (Final Office Action, page 3, citing Col. 3, lines 9-13 of *Mathur et al.).*

To support this assertion, the Examiner has initially alleged that the Inter-Process Communication (IPC) mechanism described by *Mathur et al.* teaches the claimed system handler application (Final Office Action, page 2, citing Col. 1, lines 53-38 of *Mathur et al.).* In other words, the Examiner has asserted that the IPC mechanism is a system handler application in the context of the claimed invention.

As such, the Examiner needs to at least show that the described IPC mechanism of *Mathur et al.* teaches loading a gaming program shared object in response to a change in the stored game data variables by another gaming program shared object. Instead, the Examiner has merely relied on a general description of a "reentrant" software module (Final Office Action, page 3, citing Col. 3, lines 42-47 of *Mathur et al.*).

For the Examiner's convenience, the cited section of *Mathur et al.* which is the basis of the Examiner's rejection of this claimed feature is reproduced below.

> A software module is reentrant if it can 1) be called by a first program and be partially run; 2) be interrupted by a second program (and either partially run, or run to conclusion), and 3) then be reentered by the first program without loss of information. Conversely, software that is non-reentrant cannot be interrupted and then reentered without loss of information. [*Mathur et al.,* Col. 3, lines 9-13]

Clearly, the reentrant module discussed by *Mathur et al.* does not teach or even remotely suggest this claimed feature. Accordingly, it is respectfully submitted that the Examiner's rejection is improper and should be withdrawn.

Moreover, it is respectfully submitted that *Mathur et al.* does not teach or suggest this feature. Accordingly, it is respectfully submitted that claim 1 and its dependent claims are patentable over *Mathur et al.* for at least this reason.

(b) *Mathur et al.* does not teach or suggest the claimed features of: (i) a system handler application loading a first shared object and providing Application Program Interface functions called by the first shared object, (ii) executing the first shared object, (iii) storing data variables in a nonvolatile storage, such that a second shared object or a first device handler can access the data variables in the nonvolatile storage by utilizing the look-up table of the game state storage, (iv) unloading the first shared object, (v) loading other shared objects and (vi) repeating the steps (i) through (iv) for the other shared objects (claim 71).

It is respectfully submitted that the Examiner has not addressed these claimed features. Instead, the Examiner has again relied on a general description of a "reentrant" software module (Final Office Action, page 3, citing Col. 3, lines 42-

47 of *Mathur et al.*). Clearly, the "reentrant" software module described by *Mathur et al.* does not teach or even remotely suggest these claimed features.

Moreover, it is respectfully submitted that *Mathur et al.* does not teach or suggest these claimed features. Accordingly, it is respectfully submitted that claims 71-75 are patentable over *Mathur et al.* for at least this reason.

CONCLUSION

Based on the foregoing, it is submitted that the claims are patentably distinct over the cited art of record. Additional limitations recited in the independent claims or the dependent claims are not further discussed because the limitations discussed above are sufficient to distinguish the claimed invention from the cited art. Accordingly, Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner.

Applicant hereby petitions for an extension of time which may be required to maintain the pendency of this case, and any required fee for such extension or any further fee required in connection with the filing of this Amendment is to be charged to Deposit Account No. 500388 (Order No. IGT1P369). Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER LLP

/RMahboubian/
Ramin Mahboubian
Reg. No. 44,890

P.O. Box 70250
Oakland, CA 94612-0250
(408) 255-8001